

HADOOP INSTALLATION ON MAC

Installation Prerequisites:

1. **Java Development Kit (JDK):** Hadoop is written in Java therefore we need Java to run it. I have installed JDK 8 by download it from the official Oracle website.

```
Last login: Wed Sep 18 15:05:09 on console
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % java -version
java version "1.8.0_421"
Java(TM) SE Runtime Environment (build 1.8.0_421-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.421-b09, mixed mode)
(base) gowtham@Gowthams-MacBook-Pro-2 ~ %
```

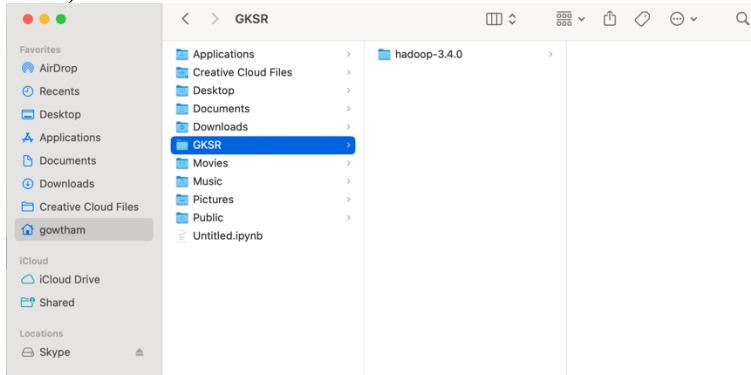
2. **SSH:** MacOS usually comes with SSH installed by default, however, to enable the feature you will need to navigate to System Preferences -> Sharing -> Remote Login. Hadoop needs SSH to transmit data between different nodes in the cluster.

```
Last login: Mon Sep 16 21:08:54 on ttys000
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/Users/gowtham/.ssh'.
Your identification has been saved in /Users/gowtham/.ssh/id_rsa
Your public key has been saved in /Users/gowtham/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:nbiYPQVUWJaGoHT+z3XYlUFDRS1Rs2f85IdoBidM7Nc gowtham@Gowthams-MacBook-Pro-2.local
The key's randomart image is:
+---[RSA 3072]----+
|   . o..*=. o*8|
|   . + .o++   .**|
Last login: Mon Sep 16 21:08:54 on ttys000
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/Users/gowtham/.ssh'.
Your identification has been saved in /Users/gowtham/.ssh/id_rsa
Your public key has been saved in /Users/gowtham/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:nbiYPQVUWJaGoHT+z3XYlUFDRS1Rs2f85IdoBidM7Nc gowtham@Gowthams-MacBook-Pro-2.local
The key's randomart image is:
+---[RSA 3072]----+
|   . o..*=. o*8|
|   . + .o++   .**|
|   . . o+ .. +B|
|   . +.=.+E*+|
|   S +.* + +|
|   + = + . .|
|   o + o    |
|           . |
|           . |
+---[SHA256]----+
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % chmod 0600 ~/.ssh/id_rsa.pub
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % ssh localhost
The authenticity of host 'localhost (::1)' can't be established.
[ED25519 key fingerprint is SHA256:XkCeQFGQPn6UIjgPVzNhMkBY90y9Y5SkAciVgYXWmZc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
Last login: Mon Sep 16 21:16:43 2024
(base) gowtham@Gowthams-MacBook-Pro-2 ~ %
Connection to localhost closed.
(base) gowtham@Gowthams-MacBook-Pro-2 ~ %
```

HADOOP INSTALLATION ON MAC

Installation of Hadoop

- Downloaded the Hadoop tar file from the link <https://hadoop.apache.org/releases.html> and extracted the Hadoop file into the directory GCSR(initials of my first name and last name).



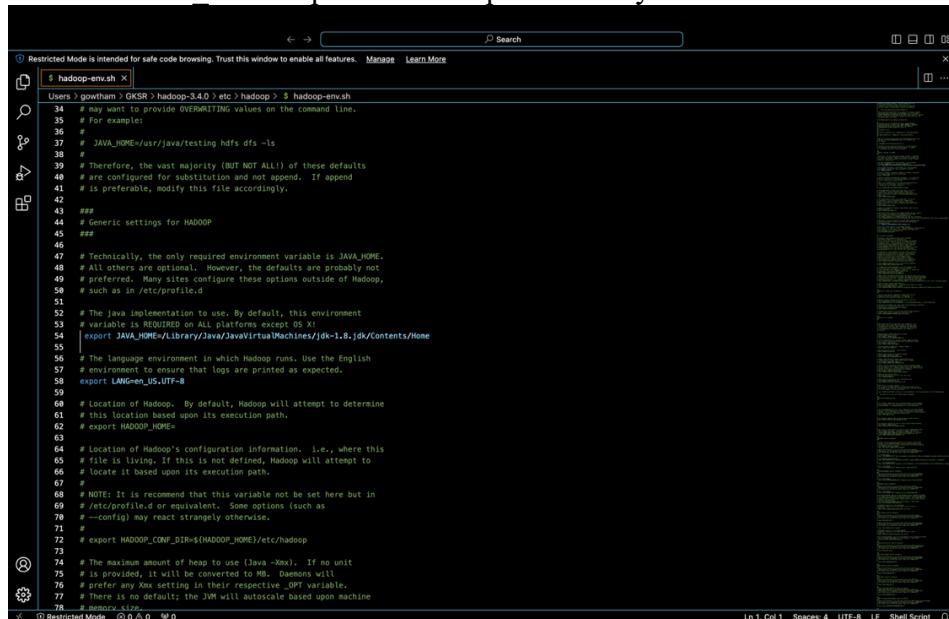
- Set up JAVA_HOME path and HADOOP_HOME path in the .zshrc file.

```
export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk-1.8.jdk/Contents/Home
export HADOOP_HOME=/Users/gowtham/GCSR/hadoop-3.4.0
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
# >>> conda initialize >>
# !! Contents within this block are managed by 'conda init' !!
__conda_setup=$( '/opt/anaconda3/bin/conda' 'shell.zsh' 'hook' 2>/dev/null )
if [ $? -eq 0 ]; then
    eval "$__conda_setup"
else
    if [ -f "/opt/anaconda3/etc/profile.d/conda.sh" ]; then
        . "/opt/anaconda3/etc/profile.d/conda.sh"
    else
        export PATH="/opt/anaconda3/bin:$PATH"
    fi
fi
unset __conda_setup
# <<< conda initialize <<<
```

- Configure the Hadoop

- hadoop-env.sh

edit the JAVA_HOME path with the path where your Java home is located.



```
34 # may want to provide OVERRIDING values on the command line.
35 # For example:
36 #
37 # JAVA_HOME=/usr/java/testing/hdfs/dfs -ls
38 #
39 # Therefore, the vast majority (BUT NOT ALL) of these defaults
40 # are configured for substitution and not append. If append
41 # is preferable, modify this file accordingly.
42 #
43 ###
44 # Generic settings for HADOOP
45 ###
46 #
47 # Technically, the only required environment variable is JAVA_HOME.
48 # All others are optional. However, the defaults are probably not
49 # preferred. Many sites configure these options outside of Hadoop,
50 # such as in /etc/profile.d
51 #
52 # The java implementation to use. By default, this environment
53 # variable is REQUIRED on ALL platforms except OS X
54 | export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk-1.8.jdk/Contents/Home
55 #
56 # The language environment in which Hadoop runs. Use the English
57 # environment to ensure that logs are printed as expected.
58 export LANG=en_US.UTF-8
59 #
60 # Location of Hadoop. By default, Hadoop will attempt to determine
61 # this location based upon its execution path.
62 # export HADOOP_HOME=
63 #
64 # Location of Hadoop's configuration information. i.e., where this
65 # file is living. If this is not defined, Hadoop will attempt to
66 # locate it based upon its execution path.
67 #
68 # NOTE: It is recommended that this variable not be set here but in
69 # /etc/profile.d or equivalent. Some options (such as
70 # --config) may react strangely otherwise.
71 #
72 # export HADOOP_CONF_DIR=$(HADOOP_HOME)/etc/hadoop
73 #
74 # The maximum amount of heap to use (Java -Xmx). If no unit
75 # is provided, it will be converted to MB. Daemons will
76 # prefer any Xmx setting in their respective _OPT variable.
77 # The default is 1024m; the default JMM will autoscale based upon machine
78 # memory size.
```

HADOOP INSTALLATION ON MAC

b. core-site.xml

Add the below property inside the configuration tag
<configuration>

```
<property>  
  
<name>fs.defaultFS</name>  
  
<value>hdfs://localhost:9000</value>  
  
</property>  
  
</configuration>
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  
<!--  
 Licensed under the Apache License, Version 2.0 (the "License");  
 you may not use this file except in compliance with the License.  
 You may obtain a copy of the License at  
  
 http://www.apache.org/licenses/LICENSE-2.0  
  
 Unless required by applicable law or agreed to in writing, software  
 distributed under the License is distributed on an "AS IS" BASIS,  
 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 See the License for the specific language governing permissions and  
 limitations under the License. See accompanying LICENSE file.  
-->  
  
<!-- Put site-specific property overrides in this file. -->  
  
<configuration>  
    <property>  
        <name>fs.defaultFS</name>  
        <value>hdfs://localhost:9000</value>  
    </property>  
</configuration>
```

c. hdfs-site.xml:

create a folder named data inside hadoop folder and create two more folders called datanode and namenode inside the data node. Then edit the hdfs-site.xml with below text.

```
<configuration>  
  
<property>  
  
<name>dfs.replication</name>  
  
<value>1</value>  
  
</property>
```

HADOOP INSTALLATION ON MAC

```
<property>
    <name>dfs.namenode.name.dir</name>
    <value>/Users/gowtham/GKSR/hadoop-3.4.0/data/namenode</value>
</property>

<property>
    <name>dfs.datanode.data.dir</name>
    <value>/Users/gowtham/GKSR/hadoop-3.4.0/data/datanode</value>
</property>

</configuration>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
    Licensed under the Apache License, Version 2.0 (the "License");
    you may not use this file except in compliance with the License.
    You may obtain a copy of the License at

        http://www.apache.org/licenses/LICENSE-2.0

    Unless required by applicable law or agreed to in writing, software
    distributed under the License is distributed on an "AS IS" BASIS,
    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
    See the License for the specific language governing permissions and
    limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>/Users/gowtham/GKSR/hadoop-3.4.0/data/namenode</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>/Users/gowtham/GKSR/hadoop-3.4.0/data/datanode</value>
    </property>
</configuration>
```

d. yarn-site.xml

Add the below properties inside the configuration tag.

```
<configuration>
```

HADOOP INSTALLATION ON MAC

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
</property>

<property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>

<property>
    <name>yarn.resourcemanager.hostname</name>
    <value>localhost</value>
</property>

<property>
    <name>yarn.acl.enable</name>
    <value>0</value>
</property>

<property>
    <name>yarn.nodemanager.env-whitelist</name>
    <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,
HADOOP_CONF_DIR,CLASSPATH_PERPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_MAPRED_HOME</value>
</property>
</configuration>
```

HADOOP INSTALLATION ON MAC

```
<?xml version="1.0"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<configuration>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
    <property>
        <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
        <value>org.apache.hadoop.mapred.ShuffleHandler</value>
    </property>
    <property>
        <name>yarn.resourcemanager.hostname</name>
        <value>localhost</value>
    </property>
    <property>
        <name>yarn.acl.enable</name>
        <value><</value>
    </property>
    <property>
        <name>yarn.nodemanager.env-whitelist</name>
        <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,CLASSPATH_PERPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_MAPRED_HOME</value>
    </property>
</configuration>
```

e. mapred-site.xml

Add the below properties inside the configuration tag of mapred-site.xml

```
<?xml version="1.0"?>
<?xmlstylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->

<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

4. Open terminal and run the command
hdfs namenode -format

The above command line is used to initialize the HDFS for the first time. It basically formats the namenode which means it creates necessary directories and metadata to store information about the files and blocks in the HDFS cluster.

HADOOP INSTALLATION ON MAC

```
Last login: Tue Sep 17 16:15:26 on ttys000
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hdfs namenode -format
WARNING: /Users/gowtham/GKSR/hadoop-3.4.0/logs does not exist. Creating.
2024-09-17 16:31:01,076 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = Gowthams-MacBook-Pro-2.local/127.0.0.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.4.0
STARTUP_MSG:   classpath = /Users/gowtham/GKSR/hadoop-3.4.0/etc/hadoop:/Users/gowtham/GKS
ebapp-9.4.53.v20231009.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/curat
KSR/hadoop-3.4.0/share/hadoop/common/lib/nimbus-jose-jwt-9.31.jar:/Users/gowtham/GKSR/had
ng-1.2.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/reload4j-1.2.22.jar:/l
/common/lib/netty-all-4.1.100.Final.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/com
8.0.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/hadoop-shaded-guava-1.2
adoop/common/lib/kerb-server-2.0.3.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/comm
ers/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/jetty-util-ajax-9.4.53.v20231009.j
4.0/share/hadoop/common/lib/kerb-identity-2.0.3.jar:/Users/gowtham/GKSR/hadoop-3.4.0/shar
ers/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/jersey-servlet-1.19.4.jar:/Users/go
n/lib/netty-transport-native-epoll-4.1.100.Final-linux-aarch_64.jar:/Users/gowtham/GKSR/h
-1.5.0.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/woodstox-core-5.4.0.j
common/lib/curator-framework-5.2.0.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/comm
/GKSR/hadoop-3.4.0/share/hadoop/common/lib/kerby-config-2.0.3.jar:/Users/gowtham/GKSR/had
/kerb-util-2.0.3.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/netty-codec
GKSR/hadoop-3.4.0/share/hadoop/common/lib/commons-io-2.14.0.jar:/Users/gowtham/GKSR/hadoo
.0.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib/javax.servlet-api-3.1.0.j
adoop/common/lib/jline-3.9.0.jar:/Users/gowtham/GKSR/hadoop-3.4.0/share/hadoop/common/lib
..
```

5. Start all Hadoop Daemons by running the command
start-all.sh

```
=====
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as gowtham in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [Gowthams-MacBook-Pro-2.local]
2024-09-17 16:31:36,741 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
```

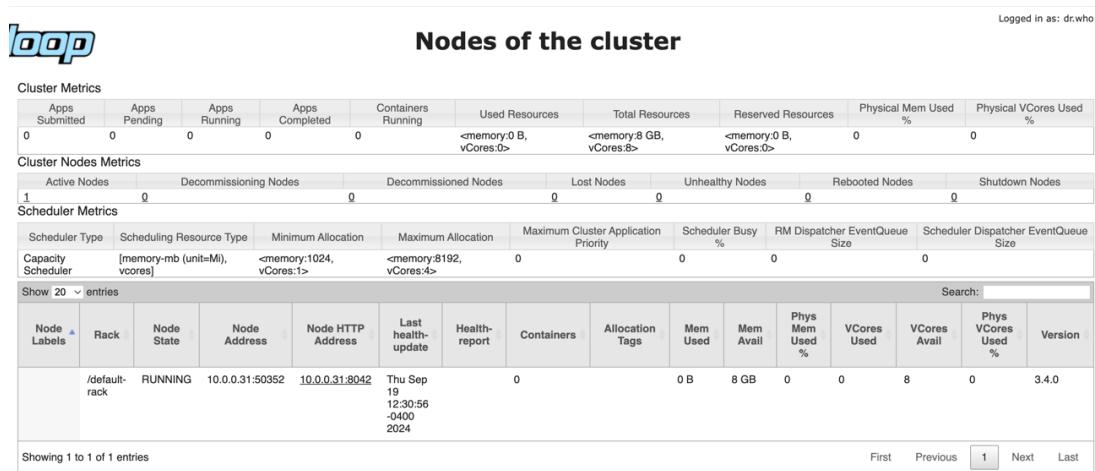
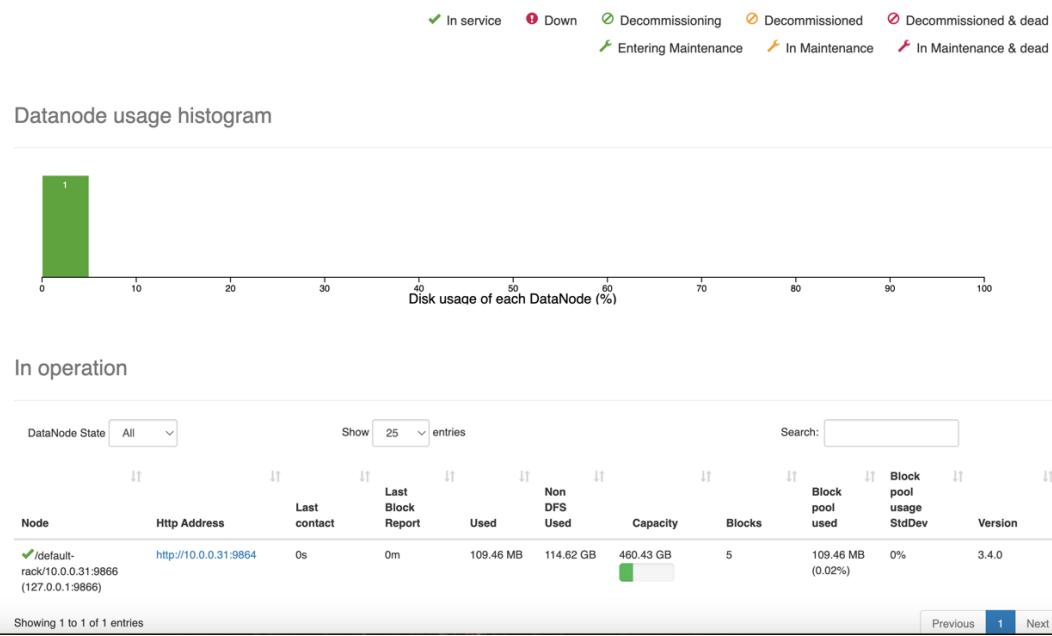
6. Verify the installation
run the command jps to verify the hadoop installation.

```
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % jps
5539 ResourceManager
5638 NodeManager
5350 SecondaryNameNode
5110 NameNode
5213 DataNode
5726 Jps
```

Open the browser and type localhost:9870, localhost:8088 to check the running nodes

HADOOP INSTALLATION ON MAC

Datanode Information



To stop all the running nodes type the command : **stop-all.sh**

Uploading PSV files:

- Download any five psv files from this link <https://www.ncei.noaa.gov/oa/global-historical-climatology-network/index.html#hourly/access/by-station/>
- First we need to make a directory in the hadoop. For that use the command: `hadoop fs -mkdir <paths>`(use the name of the directory that you want to create)
- To upload the files use the command: `hadoop fs -put <local-src> ... <HDFS_dest_path>`

The below figures shows the command lines and the files uploaded into the hadoop.

HADOOP INSTALLATION ON MAC

```
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -mkdir /user
2024-09-17 16:46:03,481 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -mkdir /user/hadoop
2024-09-17 16:46:13,979 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -mkdir /user/hadoop/GKSR
2024-09-17 16:46:52,882 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put GHCHN_ACL000BARA9_por.psv /user/hadoop/GKSR/GHCHN_ACL000BARA9_por.psv
2024-09-17 16:52:06,543 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
put: `GHCHN_ACL000BARA9_por.psv': No such file or directory
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put GHCHN_ACL000BARA9_por.psv /user/hadoop/GKSR
2024-09-17 16:52:17,558 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
put: `GHCHN_ACL000BARA9_por.psv': No such file or directory
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put /Users/gowtham/Desktop/GHCHN_ACL000BARA9_por.psv /user/hadoop/GKSR
2024-09-17 16:54:16,776 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put /Users/gowtham/Desktop/GHCHN_AC00078861_por.psv /user/hadoop/GKSR
2024-09-17 16:58:17,679 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put /Users/gowtham/Desktop/GHCHN_ACU55-00189_por.psv /user/hadoop/GKSR
2024-09-17 16:58:54,793 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put /Users/gowtham/Desktop/GHCHN_ACU55-00190_por.psv /user/hadoop/GKSR
2024-09-17 16:59:30,925 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ % hadoop fs -put /Users/gowtham/Desktop/GHCHN_ACU55-00191_por.psv /user/hadoop/GKSR
2024-09-17 17:00:17,456 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
(base) gowtham@Gowthams-MacBook-Pro-2 ~ %
```



Browse Directory

/user/hadoop/GKSR									Go!					
Show	25	entries								Search:				
	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name						
<input type="checkbox"/>	-rw-r--r--	gowtham	supergroup	46.92 MB	Sep 17 16:54	1	128 MB	GHCHN_ACL000BARA9_por.psv						
<input type="checkbox"/>	-rw-r--r--	gowtham	supergroup	52.08 MB	Sep 17 16:58	1	128 MB	GHCHN_AC00078861_por.psv						
<input type="checkbox"/>	-rw-r--r--	gowtham	supergroup	1001.24 KB	Sep 17 16:58	1	128 MB	GHCHN_ACU55-00189_por.psv						
<input type="checkbox"/>	-rw-r--r--	gowtham	supergroup	941.89 KB	Sep 17 16:59	1	128 MB	GHCHN_ACU55-00190_por.psv						
<input type="checkbox"/>	-rw-r--r--	gowtham	supergroup	7.69 MB	Sep 17 17:00	1	128 MB	GHCHN_ACU55-00191_por.psv						

Showing 1 to 5 of 5 entries

Previous 1 Next

Hadoop, 2024.

HADOOP INSTALLATION ON MAC

Learning Experience while Installing Hadoop:

Setting up and configuring a simple Hadoop cluster was part of my learning experience with Hadoop, with an eye toward knowledge of fundamental elements including the NameNode and DataNode. The distributed file system (HDFS) of Hadoop is built on these two components. While the DataNodes deal with the actual data storage, the NameNode manages the metadata and directory tree of every file in the system.

Correcting XML Configuration Files

I changed several XML configuration files throughout the installation and setup process:

- `hadoop-env.sh`: Since Hadoop is Java-based, this file needed specifying the `JAVA_HOME` path to guarantee Hadoop could find the Java Development Kit (JDK), which is required for running Hadoop.
- `core-site.xml`: I set the file system to `{hdfs://localhost:9000}` by adding the property `{name}.fs.defaultFS>`. This specifies the port via which communication takes place and the NameNode's address.
- Particularly crucial for the NameNode and DataNode configurations was `hdfs-site.xml`. I set aside directories for every one of them:
- Name Node metadata is kept at `dfs.namenode.name.dir`.
- `dfs.datanode.data.dir` indicates the directory the DataNodes keep actual data in.
- Both paths were configured for locally created folders, which are essential to guarantee the system knows where to save information and data.
- `Yarn-site.xml`: For Hadoop resource management, I turned on YARN (yet another resource negotiator). Important features including ``yarn.nodemanager.aux-services`` were included to let MapReduce workloads operate on the cluster.
- `mapred-site.xml`: This file was set up to support the Hadoop processing framework, MapReduce capability.

These modifications helped me to launch all required Hadoop daemons and effectively format the HDFS using the ``hdfs namenode -format``. At last, I confirmed the configuration by looking at running nodes on my browser.

This practical knowledge helped me to grasp the subtleties in Hadoop's architecture and settings.